***Sito Eratostenesa***

Liczby naturalne większe od 1 można podzielić na dwa zbiory: zbiór liczb pierwszych i złożonych (czyli NIE pierwszych). Każda liczba złożona jest wielokrotnością liczby pierwszej (np. 6=2⋅3, 35=5⋅7, …). Wystarczy więc ze zbioru liczb „wyrzucić” te liczby, które są wielokrotnościami i zostaną TYLKO liczby pierwsze. Na tej właśnie zasadzie działa sito Eratostenesa - przesiewa liczby wyrzucając ze zbioru ich kolejne wielokrotności. Ten algorytm świetnie nadaje się do uzyskania wszystkich liczb pierwszych z zakresu <2;n>.

*Przykład*

Mamy zbiór liczb <2;50>:

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50

 Rozpoczynamy od liczby 2. Wyrzucamy wszystkie jej wielokrotności:

 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50

 Po tej operacji w zbiorze pozostaje liczba 2 oraz liczby nieparzyste. Żadna z pozostałych liczb, oprócz 2, nie dzieli się już przez 2. Teraz to samo wykonujemy z liczbą 3. Wyrzucamy ze zbioru wszystkie jej wielokrotności:

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50  
Operację kontynuujemy z pozostałymi liczbami:

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50

I ostatecznie otrzymujemy zbiór, z którego usunięto wszystkie liczby złożone:

**2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50**

Pozostałe w zbiorze liczby są liczbami pierwszymi: 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47.

Sito Eratostenesa to bardzo szybki algorytm, który w danym zbiorze kolejnych liczb naturalnych wyznacza nam wszystkie liczby pierwsze. Szybkość jest tutaj okupiona zajętą pamięcią - algorytm wyznaczania liczb pierwszych przez sprawdzanie podzielności miał bardzo małe wymagania pamięciowe, lecz działał wolno. **Sito Eratostenesa działa bardzo szybko, lecz ma duże wymagania pamięciowe.** Coś za coś.

**Algorytm sita Eratostenesa**

Realizując ten algorytm na komputerze musimy wybrać odpowiedni sposób na pamiętanie zbioru liczb, z których można usuwać w prosty sposób wielokrotności liczb. Użyjemy do tego celu tablicy. Poszczególne elementy tej tablicy będą reprezentowały liczby zbioru. Natomiast zawartość tych elementów będzie informacją, czy dany element pozostaje w zbiorze, czy też został z niego usunięty.

Najlepszym typem danych dla tablicy będzie typ logiczny bool. Jeśli element tablicy będzie miał wartość true, to znaczy, że reprezentowana przez jego indeks liczba znajduje się w zbiorze. Jeśli będzie miał wartość false, to ta liczba została ze zbioru usunięta. Na początku algorytmu umieszczamy true we wszystkich elementach tablicy - odpowiada to pełnemu zbiorowi liczb. Następnie do elementów, których indeksy są wielokrotnościami początkowych liczb, będziemy wpisywać false. Na końcu algorytmu wystarczy przeglądnąć tablicę i wyprowadzić indeksy tych elementów, które zachowały wartość true - nie są one wielokrotnościami żadnych wcześniejszych liczb, a zatem są liczbami pierwszymi.

**DANE:**

   n - określa górny kraniec przedziału <2,n>, w którym poszukujemy liczb pierwszych  
**WYNIK:**

    Liczby pierwsze z przedziału <2,n>

**Zmienne pomocnicze:**

T[ ] - tablica o elementach logicznych, których indeksy obejmują przedział <2,n>

**Zadanie:** Napisz program, znajdujący liczby pierwsze z przedziału <2,n> za pomocą sita Eratostenesa.